Analyse de la tenue sismique des grands réservoirs métalliques

Résumé :

Ce document a pour objectif la description des différentes étapes à mettre en œuvre dans *Code_Aster* pour réaliser la simulation numérique de la tenue au séisme de grands réservoirs métalliques. Ces structures métalliques minces peuvent alors présenter deux modes de ruine privilégiés : rupture des ancrages ou flambage des viroles.

La première approche est basée sur une méthode réglementaire de type push-over. On peut, en complément, utiliser un critère de stabilité non linéaire (*cf.* cas-tests SSLL105 et SSNL126).

La seconde approche est la modélisation transitoire directe avec prise en compte complète du domaine fluide au travers d'une approche couplée fluide-structure en grands déplacements (modélisation semblable à celle du cas-test FDNV100).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Table des matières

1 Introduction....3
2 Présentation de la problématique....4
2.1 Géométrie de la structure....4
2.2 Matériaux employés....8
3 Approche quasistatique push-over....9
3.1 Chargements imposés....9
3.2 Méthode de résolution quasistatique monotone non linéaire....11
3.3 Remarques sur les calculs et post-traitements....13
3.4 Modélisation fine des ancrages : soulèvement....14
3.5 Utilisation du critère de stabilité non linéaire....19
3.6 Pilotage du chargement....21
4 Approche transitoire couplée....21
4.1 Problème de référence....21
4.2 Modélisation couplée fluide-structure dans le Code_Aster....23
4.3 Conditions aux limites....26
4.4 Conditions initiales....28
4.5 Résolution transitoire....30
5 Bibliographie....32

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 Introduction

Cette documentation présente les différentes méthodologies numériques développées pour simuler et analyser la tenue sismique de grands réservoirs métalliques.
Il s'agit donc ici de présenter la mise en œuvre et l'enchaînement de divers opérateurs du *Code_Aster* pour mener à bien l'étude de ce type de composants, suivant différentes hypothèses de modélisation qui sont principalement dictées par des considérations réglementaires.
Ces réservoirs, qui sont des structures métalliques minces, peuvent présenter deux modes de ruine privilégiés : rupture des ancrages ou flambage des viroles.

La première méthodologie est basée sur une méthode réglementaire de type push-over ([bib1], [bib2], [bib3], [bib4]). Le réservoir est soumis à une pression interne imposée, spatialement variable, croissante. La résolution se fait en quasi-statique et le domaine fluide n'est pas modélisé directement : son influence sur la paroi est transcrite par un champ de pression imposé particulier. Les non linéarités sont de type géométrique et comportementale (plasticité).
Au cours du calcul incrémental, on fait croître cette pression jusqu'à obtenir la charge ultime qui correspond au flambage de la structure.
Afin de modéliser plus finement les ancrages, on peut introduire le soulèvement au niveau des ancrages boulonnés avec le sol [bib5].
On peut aussi augmenter la tenue mécanique au flambage en rajoutant un renfort en fibre de carbone sur les viroles. Sa modélisation est présentée dans ce document.

La seconde approche est la modélisation transitoire directe avec prise en compte complète du domaine fluide au travers d'une approche couplée fluide-structure en grands déplacements [bib6]. Cette modélisation, plus fine que la précédente, vient compléter les approches réglementaires, en particulier en permettant de mieux cerner les limites de leur domaine de validité, principalement vis-à-vis des grandes non linéarités globales comme les grands déplacements.
Cependant, l'usage pratique de ces approches transitoires est limité par le surcoût numérique qu'elles induisent (rapport de l'ordre de 10).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Présentation de la problématique

2.1 Géométrie de la structure

Le problème, dans le cas d'une bâche type, peut être représenté géométriquement ainsi [bib2] :

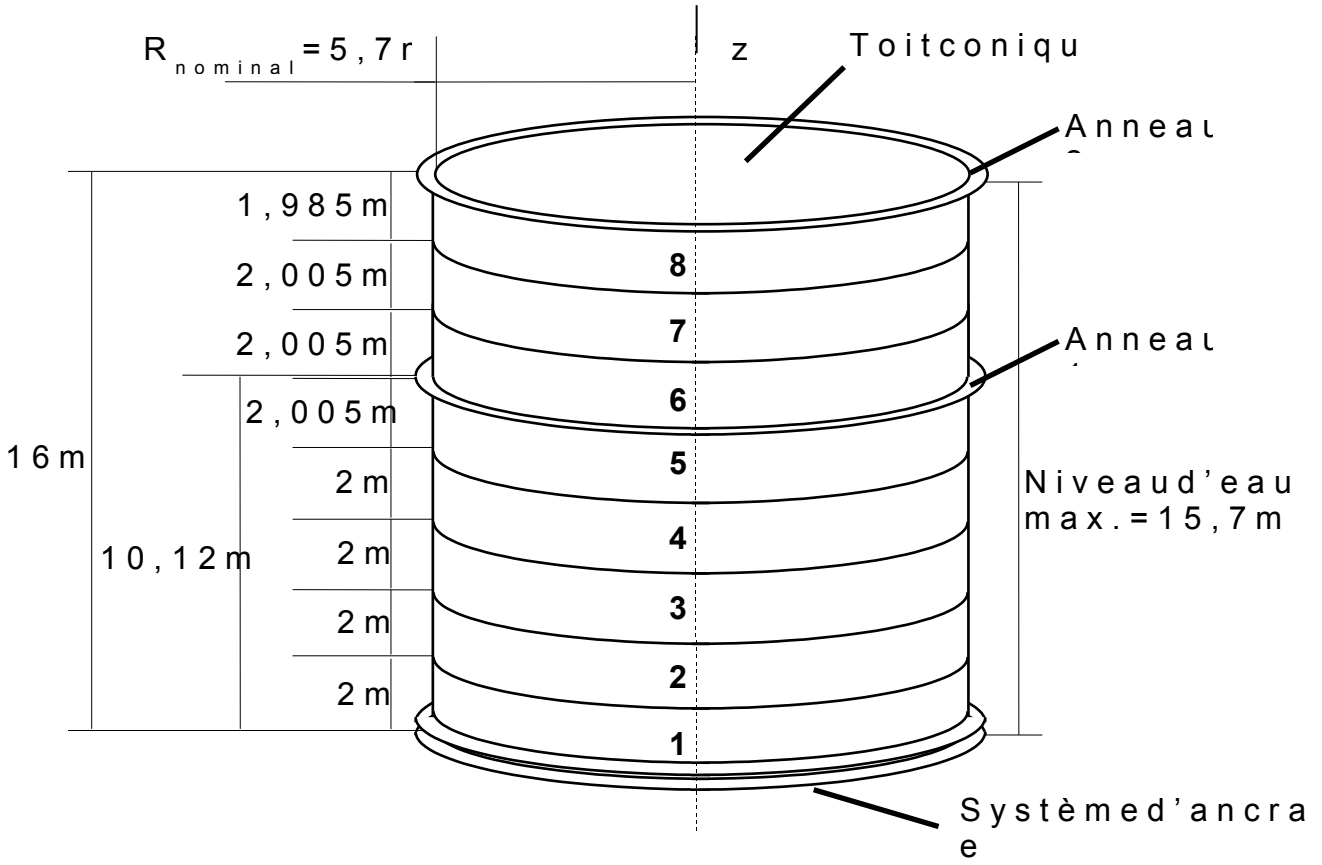

Figure 2.1-a : Représentation schématique d'une bâche réelle

La bâche présentée est composée de 8 viroles cylindriques d'environ 2 m de hauteur chacune, qui sont soudées entres elles pour former la paroi cylindrique du réservoir. L'épaisseur, constante par virole, va en décroissant quand on passe d'une virole à celle qui la surplombe. Cette disposition particulière permet d'optimiser l'épaisseur des viroles, en fonction de la pression hydrostatique exercée par le fluide contenu et qui varie linéairement avec la profondeur.

Etant données les très faibles épaisseurs des viroles (environ 4 à 5 mm), on choisit de modéliser les parois métalliques par des coques minces volumiques (COQUE_3D).
Ces éléments ont pour support géométrique un quadrilatère à 9 nœuds.
Certains mailleurs ne savent pas générer ce type d'élément mais proposent des maillages avec des quadrilatères à 8 nœuds plus classiques. On vient donc définir les nœuds milieux dans le *Code_Aster* par la commande suivante :

```
MAILLA2=CREA_MAILLAGE(MAILLAGE=MAILLA01,
                      MODI_MAILLE=_F(GROUP_MA='RESERVOIR',
                                     OPTION='QUAD8_9',
                                     PREF_NOEUD='NSQ',
                                     PREF_NUME=1,),);
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

On s'assure ensuite que les normales sont correctement orientées :

```
MAILLA2 = MODI_MAILLAGE( reuse = MAILLA2 , MAILLAGE = MAILLA2 ,
                         ORIE_NORM_COQUE = _F( GROUP_MA =
                                               ( 'VIROLE', 'TFC2' ) ,
                                              VECT_NORM = (0.,-1.,0.) ,
                                              GROUP_NO = 'OY' , ) ,
                         MODELE = MODELE0 , INFO = 2 , )
```

On peut alors définir le modèle MODELE définitif basé sur les coques volumiques.

On définit les caractéristiques géométriques propres aux éléments de structures employés (COQUE_3D). Virole par virole on donne les épaisseurs et l'orientation des repères. Il faut particulièrement être vigilant à la cohérence entre les axes des repères et les caractéristiques matériaux quand on est dans un cas orthotrope (cas du renfort). De même pour ce renfort, on peut se servir de l'excentrement pour tenir compte de sa disposition réelle sur l'extérieur de la virole métallique (sa surface moyenne ne peut donc être confondue avec celle de la virole le supportant) :

```
CARAELEM = AFFE_CARA_ELEM ( MODELE = MODELE ,
                            COQUE = ( _F ( GROUP_MA = ( 'SURF0' , ) ,
                                           EPAIS = 7.13E-3 ,
                                           ANGL_REP = ( 0. , 90. ) ,
                                           A_CIS = 0.8333 ,
                                           COEF_RIGI_DRZ = 1.E-05 ,
                                           EXCENTREMENT = 0. ,
                                           INER_ROTA = 'OUI' ,
                                           MODI_METRIQUE = 'OUI' , ) ,
…
                                      _F ( GROUP_MA = ( 'ANNEAU' , ) ,
                                           EPAIS = 1.E-2 ,
                                           ANGL_REP = ( 0. , 0. ) ,
                                           A_CIS = 0.8333 ,
                                           COEF_RIGI_DRZ = 1.E-05 ,
                                           EXCENTREMENT = 0. ,
                                           INER_ROTA = 'OUI' ,
                                           MODI_METRIQUE = 'OUI' , ) ,
                                      _F ( GROUP_MA = ( 'TFC2' , ) ,
                                           EPAIS = ( 1. * 1.72E-3 ) ,
                                           ANGL_REP = ( 0. , 90. ) ,
                                           A_CIS = 0.8333 ,
                                           EXCENTREMENT = -4.425E-3 ,
                                           INER_ROTA = 'OUI' ,
                                           MODI_METRIQUE = 'OUI' ,),),)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2.2 Matériaux employés

La tôlerie est en acier inoxydable austénitique A240 type 304L et la boulonnerie sera prise en compte comme étant en acier A42.
Pour l'acier A240, différentes lois élastoplastiques avec écrouissage isotrope non linéaire (sous forme de courbes de tractions) sont proposées. La définition de l'écrouissage à partir de courbes de tractions, sans information sur la décharge, est bien adaptée aux applications où le chargement est monotone croissant, ce que l'on va supposer dans notre cas (*cf.* § 3.1) :

```
A5=DEFI_FONCTION(NOM_PARA='EPSI',
                 VALE=( 0.00097 , 191820000. , 0.00116 , 221740000. ,
                        0.00138 , 247020000. , 0.00167 , 267090000. ,
                        0.00207 , 285920000. , 0.0029 , 304530000. ,
                        0.00365 , 314300000. , 0.00468 , 322090000. ) ,
                 PROL_DROITE='LINEAIRE',
                 PROL_GAUCHE='LINEAIRE',);
#
MAA240 = DEFI_MATERIAU ( ELAS = _F ( E = 1.98E11 ,
                                     NU = 0.3 ,
                                     RHO = 7900. , ) ,
                         TRACTION = _F ( SIGM = A5 , ) , )
```

Cette relation de comportement est du type 'VMIS_ISOT_TRAC' dans l'opérateur de résolution non linéaire employé.

Le comportement plastique de la nuance A42 ne sera pas pris en compte pour commencer. En effet, la limite élastique est plus élevée que pour le 304L et les pièces constituées de ce matériau sont plus massives, ce qui engendre des niveaux de contraintes plus faibles. L'hypothèse de comportement purement élastique isotrope est donc admissible pour le matériau correspondant, nommé MAA42.

Dans le cas où l'on rajoute un renfort en tissu de fibres de carbone (TFC), on le modélise par un matériau élastique orthotrope (le seuil de plasticité n'est jamais atteint pour nos chargements) :

```
MATFC = DEFI_MATERIAU ( ELAS_ORTH = _F ( E_L = 1.E10 ,
                                         E_T = 1.E12 ,
                                         G_LT = 3.E9 ,
                                         NU_LT = 0.3 ,
                                         RHO = 2500. , ) , )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Approche quasistatique push-over

3.1 Chargements imposés

Le chargement imposé est d'origine réglementaire (EC-8), quasistatique [bib1]. On représente les effets d'un séisme sur la structure par un champ de pression variable imposé sur la face interne des viroles. La valeur en chaque point est fonction de la coordonnée courante et croît linéairement avec le temps. Cette évolution monotone en temps est caractéristique des méthodes dites push-over, au sens EC-8, dont l'objectif est de simuler par un calcul quasistatique la réponse à une sollicitation sismique, et donc de nature physique dynamique transitoire. Les effets dynamiques, comme l'inertie et les efforts générés par le fluide en ballottement sont remplacés par cette distribution de pression imposée. Le problème à traiter ne fait intervenir qu'une modélisation de la structure, sans modélisation du domaine fluide.
Les méthodes push-over ont bien évidemment été construites et justifiées en faisant des hypothèses fortes de linéarisation du problème (petits déplacements, comportement élastique, flambement d'Euler…). Ci-dessous sont données les définitions des champs de pressions imposés [bib2] :

```
# ----------------------------------------------------------------------
#    DEFINITION DES CHAMPS DE PRESSION  (PA)
#
#       PH(Z)       : PRESSION HYDROSTATIQUE
#       PIF(Z,TETA) : PRESSION IMPULSIVE FLEXIBLE
#       PIR(Z,TETA) : PRESSION IMPULSIVE RIGIDE
#       PV(Z)       : PRESSION VERTICALE
#
#       PIF(Z,TETA) = PIF(Z)*COS(TETA)
#       PIR(Z,TETA) = PIR(Z)*COS(TETA)
#
#P(Z,TETA)=PH(Z)+INSTANT*( PIF(Z,TETA) + PIR(Z,TETA) +/- 0.4*PV(Z) )
# ----------------------------------------------------------------------
PHZ = DEFI_FONCTION (  NOM_PARA = 'Z' ,
…
                       VERIF = 'CROISSANT' ,  )
PIFZ = DEFI_FONCTION (  NOM_PARA = 'Z' ,
…
                         VERIF = 'CROISSANT' ,  )
PIRZ = DEFI_FONCTION (  NOM_PARA = 'Z' ,
…
                        VERIF = 'CROISSANT' , )
PVZ = DEFI_FONCTION (  NOM_PARA = 'Z' ,
…
                       VERIF = 'CROISSANT' , )
# ----------------------------------------------------------------------
#       P(X,Y,Z)    : PRESSION TOTALE
#P(Z,TETA)=PH(Z)+INSTANT*( PIF(Z,TETA) + PIR(Z,TETA) +/- 0.4*PV(Z) )
# ----------------------------------------------------------------------
COSTE=FORMULE(NOM_PARA=('X','Y'),VALE='X/SQRT((X*X)+(Y*Y))')
PH=FORMULE(NOM_PARA=('X','Y','Z'),VALE='PHZ(Z)')
PIF=FORMULE(NOM_PARA=('X','Y','Z'),VALE='PIFZ(Z)*COSTE(X,Y)')
PIR=FORMULE(NOM_PARA=('X','Y','Z'),VALE='PIRZ(Z)*COSTE(X,Y)')
PV=FORMULE(NOM_PARA=('X','Y','Z'),VALE='PVZ(Z)')
PS1=FORMULE(NOM_PARA=('X','Y','Z'),
            VALE='PIF(X,Y,Z)+PIR(X,Y,Z)+(0.4*PV(X,Y,Z))')
PS2=FORMULE(NOM_PARA=('X','Y','Z')
            VALE='PIF(X,Y,Z)+PIR(X,Y,Z)-(0.4*PV(X,Y,Z))')
```

On peut alors définir les chargement mécaniques correspondants (pesanteur et pressions suiveuses) :

```
PESA = AFFE_CHAR_MECA( MODELE = MODELE ,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                          PESANTEUR = ( 9.81 , 0. , 0. , -1. ) , )
#
PRESPH = AFFE_CHAR_MECA_F( MODELE = MODELE ,
       FORCE_COQUE = _F ( GROUP_MA = 'VIROLE',),PRES = PH, PLAN = 'INF',),)
#
PRESPS1 = AFFE_CHAR_MECA_F( MODELE = MODELE ,
                            FORCE_COQUE = _F ( GROUP_MA = 'VIROLE' ,  ) ,
                                               PRES = PS1 ,
                                               PLAN = 'INF' , ) , )
#
PRESPS2 = AFFE_CHAR_MECA_F( MODELE = MODELE ,
                            FORCE_COQUE = _F ( GROUP_MA =
                                               ( 'VIROLE' ,  ) ,
                                               PRES = PS2 ,
                                               PLAN = 'INF' , ) , )
```

Ces champs de pressions sont multipliés par une fonction croissante linéaire du temps :

```
FONCMUL = DEFI_FONCTION( NOM_PARA = 'INST' , = ( 0. , 0. , 3. , 3. ) , )
```

La liaison avec le sol est considérée ici comme étant complète (groupe de noeuds `'BASE'`). On définit aussi les conditions de symétrie (groupe de nœuds `'SYMETRIE'`) puisqu'on ne maille qu'un demi réservoir :

```
CONDLIM = AFFE_CHAR_MECA( MODELE = MODELE ,
                          DDL_IMPO = ( _F ( GROUP_NO = 'BASE' ,
                                            DX = 0., DY = 0., DZ = 0.,
                                            DRX = 0., DRY = 0., DRZ = 0.,) ,
                                           _F ( GROUP_NO = 'SYMETRIE' ,
                                            DY = 0., DRX = 0., DRZ = 0.,),) , )
```

3.2 Méthode de résolution quasistatique monotone non linéaire

On veut résoudre un problème d'évolution quasistatique non linéaire. On va donc utiliser l'opérateur `STAT_NON_LINE` ([U4.51.03], [R5.03.01]). Le chargement imposé sera construit avec la pression `PRESPS1` par exemple :

```
RESU = STAT_NON_LINE( MODELE = MODELE ,
                      CHAM_MATER = CHMAT ,
                      CARA_ELEM = CARAELEM ,
                      EXCIT = ( _F ( CHARGE = CONDLIM ,
                                     TYPE_CHARGE = 'FIXE_CSTE' , ) ,
                                _F ( CHARGE = PESA ,
                                     TYPE_CHARGE = 'FIXE_CSTE' , ) ,
                                _F ( CHARGE = PRESPH ,
                                     FONC_MULT = FONCMUL ,
                                     TYPE_CHARGE = 'SUIV' , ) ,
                                _F ( CHARGE = PRESPS1 ,
                                     FONC_MULT = FONCMUL ,
                                     TYPE_CHARGE = 'SUIV' , ) , ) ,
                      COMP_INCR = ( _F ( RELATION = 'ELAS' ,
                                         COQUE_NCOU = 1 ,
                                         DEFORMATION = 'GREEN_GR' ,
                                         GROUP_MA = ( 'SURF2' ,
                                         'ANNEAU' , 'TFC2' ) , )  ,
                                    _F ( RELATION = 'VMIS_ISOT_TRAC' ,
                                         COQUE_NCOU = 1,
                                         DEFORMATION = 'GREEN_GR' ,
                                         GROUP_MA =
                                      ( 'SURF0' , 'SURF1' , ) , )  , ) ,
                      INCREMENT = _F ( LIST_INST = L_INST1 ,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    SUBD_PAS = 4 ,
                                   SUBD_PAS_MINI = 1.E-4 ,
                                   SUBD_METHODE='UNIFORME' ) ,
…
                                        )
```

3.3 Remarques sur les calculs et post-traitements

Dans l'exemple présenté ci-dessus on utilise un algorithme de Newton : on résout avec l'opérateur tangent global (raideur) réactualisé à chaque itération. Dans le cas où le problème est bien posé (suffisamment « régulier »), il est connu que ce type d'algorithme offre la meilleure convergence. Donc, dans notre cas, tant que l'opérateur tangent global est « bien » défini (loin d'être singulier), le calcul va bien se dérouler avec une convergence rapide. Quand le niveau de charge imposé va se rapprocher de la charge ultime, la structure va alors devenir instable au sens du flambage [R7.05.01], ce qui se traduit par la tendance de l'opérateur tangent global à devenir singulier. La perte de stabilité par point limite est en fait la perte d'unicité de la solution, soit donc la singularité de l'opérateur de résolution. Au voisinage du point limite l'algorithme de Newton convergera moins bien, d'où la nécessité d'imposer des incréments de temps plus petits et une augmentation du nombre d'itérations sur le résidu en équilibre.
D'une manière générale, plus l'on va s'approcher de la charge ultime, plus le pas de temps devra être petit. Malgré cela les risques d'arrêt du calcul sur non convergence sont importants, d'où l'obligation de mener le calcul suivant plusieurs poursuites successives.
Il est néanmoins possible d'améliorer cette convergence en changeant en cours de calcul d'algorithme et de basculer sur un quasi-Newton. Pour cela, il suffit de résoudre sur l'opérateur tangent que l'on ne réactualise qu'à chaque pas (entre deux itération on garde le même), et si cela est insuffisant, on peut alors résoudre avec l'opérateur élastique qui, lui, ne peut devenir singulier. Ce choix renforce la robustesse de l'algorithme en termes de convergence, mais il augmente, parfois considérablement, le nombre d'itérations (et/ou de pas) nécessaires pour obtenir la solution.

Pour notre type d'étude on peut distinguer deux types de quantités d'intérêt pour le post-traitement. D'une part une grandeur apte à traduire le flambement de la structure et donc à faire apparaître la charge ultime. Pour cela on peut tracer le niveau de pression (coefficient multiplicateur) fonction du déplacement d'un point situé au sommet du réservoir (à l'extrémité de la génératrice la plus soumise à de la compression). D'autre part un indicateur plus local de l'apparition de la plasticité : l'isovaleur de la déformation plastique cumulée, à chaque instant de calcul.

Ces deux post-traitements ne présentent aucune difficulté particulière dans le *Code_Aster*.

Les modèles éléments finis mis en œuvre comportaient entre 55000 et 110000 ddl, menant à un temps de calcul total de l'ordre de 5 h à 10 h sur Compaq AlphaServer. Ces modèles se basaient sur une représentation simplifiée des ancrages : la géométrie des renforts et goussets est maillée finement, mais le boulonnement n'est pas présent et il est remplacé par une condition d'encastrement sur tous les nœuds d'altitude 0 m. Une modélisation plus réaliste des ancrages, avec décollement aurait donc entraîné une taille de problème global sensiblement plus grande.

La méthodologie de calcul présentée ici, dont l'objectif est d'étudier la réponse quasistatique en flambage [R7.05.01] d'un réservoir est très proche du cadre de la notice de calcul au flambage [U2.08.04]. Cette documentation présente les analyses de stabilité linéaire (au sens d'Euler) et la mise au point d'un calcul non linéaire de type push-over pour obtenir la charge ultime.

3.4 Modélisation fine des ancrages : soulèvement

Le type de réservoirs étudiés ici sont boulonnés au sol [bib5]. Ces boulons (ou tirants) traversent des brides renforcées soudées en base de virole). Pour plus de rapidité de calcul nous présentons par la suite la méthode mise en œuvre sur un maillage simplifié, mais avec des ancrages réalistes : le réservoir est fixé au sol par 18 tirants.

Dans le *Code_Aster*, plusieurs modélisations du contact sont disponibles.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

La modélisation de la géométrie de la zone de contact peut être surfacique (problème 3D), linéique (problème 2D) ou constituée d'éléments discrets (mot-clés DIS_CHOC dans STAT_NON_LINE et DIS_CONTACT pour la loi matériau).
Le contact lui-même peut être traité, soit de façon nodale par pénalisation ou par la méthode des lagrangiens, avec contraintes actives ou non, soit de façon continue par la méthode des lagrangiens augmentés.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

La méthode la plus simple est celle de la pénalisation, qui, pour notre problème, présente aussi l'avantage de pouvoir tenir compte du rôle du joint d'étanchéité sans avoir à le mailler séparément. Pour un calcul quasistatique, la pénalisation n'est pas numériquement problématique, comme en dynamique où cette technique peut engendrer des perturbations hautes fréquences (liées à la raideur de pénalisation). De plus, pour des éléments de structure, on peut caler la raideur de pénalisation de manière à approcher au mieux la raideur de contact de la pièce qui est en réalité massive.
Les autres méthodes sont cependant plus rigoureuses car elles n'engendrent pas d'interpénétration.
Pour l'étude du décollement de la bâche, nous allons donc utiliser une méthode de pénalisation, qui présente le meilleur rapport qualité de modélisation / coût de calcul sur notre cas précis.
Étant donnée l'option prise d'obliger tous les nœuds à ne pouvoir se déplacer que verticalement, on peut profiter de cet appariement total pour se contenter d'utiliser des éléments discrets pour le contact. En effet, il n'est pas nécessaire de faire du réappariement, et donc, les modélisations générales surfaciques maître-esclave des zones de contact sont inutilement lourdes. On va donc placer un tapis d'éléments discrets de contact sous la frette basse. On aura alors un élément discret (DIS_T sur maille SEG2) sous chaque nœud d'altitude 0.
Si la solution calculée s'avère trop éloignée de la solution de référence expérimentale, il sera nécessaire de quantifier l'influence de la condition de déplacement uniquement vertical. Il suffira de relancer des calculs sans cette hypothèse cinématique, avec, si les déplacements horizontaux sont grands, une méthode de gestion du contact avec réappariement des nœuds des surfaces concernées.

Remarques

Rigoureusement, il faudrait doter tous les nœuds de la virole de base d'un élément discret de contact. Or, si l'on utilise des éléments de type COQUE_3D pour les frettes, on ne peut placer d'élément de contact au niveau des nœuds au centre des éléments. En effet, ce nœud présente la particularité de ne porter que des degrés de liberté de type rotation. La condition de contact, qui porte sur le déplacement normal à la face, ne peut donc être exprimée en ces nœuds. Les éléments de contact, ainsi que la condition cinématique de déplacement vertical des nœuds de base ne doivent donc être portées que par les nœuds sommets ou milieux des arêtes.

La modélisation du contact par éléments discrets permet, dans le Code_Aster, de définir directement la précharge de ces éléments. On peut ainsi représenter la précontrainte des tirants, sans avoir à prédéformer ces éléments, comme on devrait le faire si l'on avait choisi une autre modélisation du contact s'appuyant sur les surfaces en vis-à-vis.

Ci-dessous, nous présentons la position des éléments discrets de contact pénalisé (en gris clair) et des 9 ressorts modélisant les tirants (en noir). Le renforcement en base de virole est indiqué en pointillés :

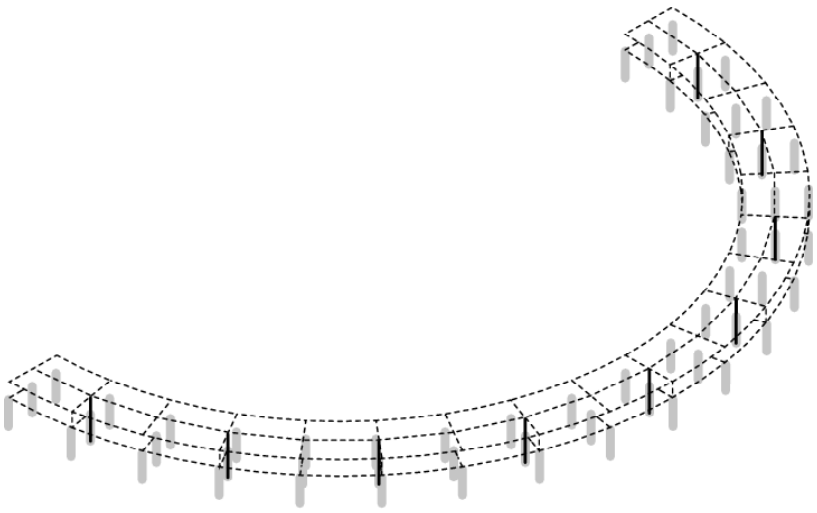

Figure 3.4-a : Disposition des éléments discrets pour la liaison avec la dalle

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Ces éléments discrets pour les ancrages sont groupés dans 2 groupes de mailles :
•RESSC : éléments de contact en gris clair (un élément SEG2 par nœud de la bride basse, soit 57 SEG2),
•RESS : tirants en noir (9 ressorts par éléments SEG2).

On introduit donc un matériau supplémentaire (MATRES) qui correspond aux éléments de contact et qui permet de prendre en compte la précontrainte par le serrage des écrous sur les tirants (on définit aussi la distance associée à la hauteur entre brides avec DIST_1 et DIST_2) :

```
Eboulon = 2.E11 ;
Dboulon = 0.026 ;
Sboulon = 3.14159 * Dboulon * Dboulon / 4 ;
kressort = Eboulon * Sboulon ;
prechar = DEFI_CONSTANTE ( VALE= -1000.,)
kpenal = kressort * 100.
MATRES = DEFI_MATERIAU( DIS_CONTACT = _F( RIGI_NOR = kpenal ,
                                          EFFO_N_INIT = prechar ,
                                    DIST_1 = 0.05, DIST_2 = 0.05,) , );
```

On complète l'affectation matériau avec ces éléments discrets sur le groupe de mailles RESSC :

```
CHMAT = AFFE_MATERIAU( MAILLAGE = MAILLA2 ,
                       AFFE = ( _F ( GROUP_MA = 'VIROLE'  , ) ,
                                   MATER = MAA240 , ) ,
                                 _F ( GROUP_MA = ( 'ANNET' , ) ,
                               MATER = MABID , )  ,
                                 _F ( GROUP_MA = ( 'ANNEA' , ) ,
                                   MATER = MAA240 , )  ,
                                 _F ( GROUP_MA = ( 'RESSC' , ) ,
                                   MATER = MATRES , ) , ) , )
```

Les caractéristiques élémentaires des nouveaux éléments discrets sont définies ainsi :

```
CARAELEM = AFFE_CARA_ELEM( MODELE = MODELE ,
                           DISCRET = (_F( GROUP_MA = 'RESSC' ,
                                  REPERE = 'LOCAL', CARA = 'K_T_D_L',
                                         VALE = ( 10. , 0.0 , 0.0 ,) ,) ,
                                       _F( GROUP_MA = 'RESS' ,
                                         REPERE = 'LOCAL', CARA = 'K_T_D_L',
                                         VALE = ( kressort , 0., 0.,),),),)
```

Les ressorts qui modélisent les tirants (GROUP_MA RESS) ont bien la raideur équivalente kressort.

Il reste à modifier les conditions aux limites en déplacement pour les ancrages : on n'autorise que les déplacements suivant la verticale des nœuds de la face inférieure de la bride basse (nœuds initialement en contact avec le sol).

L'opérateur STAT_NON_LINE ([U4.51.03], [R5.03.01]) voit aussi ses arguments impactés par l'introduction du soulèvement en base (mot-clé DIS_CHOC pour le groupe de maille RESSC) :

```
RESU = STAT_NON_LINE( MODELE = MODELE , _MATER = CHMAT ,
…
                      C OMP_INCR = ( _F( RELATION = 'ELAS' ,
                                         DEFORMATION = 'GREEN_GR' ,
                                         GROUP_MA = 'ANNEAU' , )  ,
                                     _F( RELATION = 'ELAS' ,
                                         GROUP_MA = 'RESS' , )  ,
                                     _F( RELATION = 'DIS_CHOC' ,
                                         GROUP_MA = 'RESSC' , ) ,
                                  _F( RELATION = 'VMIS_ISOT_TRAC' ,
                                         DEFORMATION = 'GREEN_GR' ,
                                         GROUP_MA =  ( 'SURF0' , 'ANNEA' ,
                                          'SURF1' , 'SURF2' ) , ) , ) ,
…
                                    );
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Le reste du fichier de commande est inchangé.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.5 Utilisation du critère de stabilité non linéaire

On peut également utiliser un critère de stabilité basé sur la matrice tangente : on a une instabilité si la matrice de raideur tangente devient singulière, *i.e.* si au moins une de ses valeurs propres s'annule. On résout alors le problème aux valeurs propres suivant, écrit en grands déplacements (écriture en lagrangien avec le tenseur de déformation de Green-Lagrange) [R7.05.01] :

$$\left(\mathbf{K}_T + \lambda I_d\right)\ \mathbf{x} = 0 \Leftrightarrow \mathbf{K}_T\mathbf{x} = \lambda I_d \mathbf{x}$$

Avec
- $\mathbf{K}_T = \mathbf{K} + \mathbf{K}^{\mathrm{L}}(\underline{u}) + \mathbf{K}^{\mathrm{Q}}(\underline{u}) + \mathbf{K}(\boldsymbol{\pi})$: matrice de rigidité tangente,
- $\mathbf{K}^{\mathrm{L}}(\underline{u})$: partie linéaire en $\underline{u}$ de la matrice $\mathbf{K}_T$,
- $\mathbf{K}^{\mathrm{Q}}(\underline{u})$: partie quadratique en $\underline{u}$ de la matrice $\mathbf{K}_T$,
- $\mathbf{K}(\boldsymbol{\pi})$: matrice de raideur géométrique,
- $\boldsymbol{\pi}$: le tenseur de Piola Kirchhoff II,
- I_d : matrice identité,
- λ : valeur propre.

La documentation [R7.05.01] présente ces analyses de stabilité plus en détail.

Remarque

Lorsque les déplacements sont petits, on a simplement $\mathbf{K}(\boldsymbol{\pi}) = \mathbf{K}(\boldsymbol{\sigma}) = \mathbf{K}_{\mathrm{G}}$ *et les matrices* $\mathbf{K}^{\mathrm{L}}$ *et* $\mathbf{K}^{\mathrm{Q}}$ *peuvent être négligées.*

Si l'on veut se servir du critère de stabilité, il suffit de rajouter la ligne suivante parmi les arguments de STAT_NON_LINE ([U4.51.03], [R5.03.01]) :

```
CRIT_FLAMB=_F(NB_FREQ=1,),
```

On cherche alors la première valeur propre de l'opérateur global tangent de notre système.

Si au cours du calcul on observe que cette valeur propre diminue, voire change de signe, cela signifie que l'on s'est approché de la première charge critique et qu'ensuite on l'a même dépassée.

Le nombre de valeurs propres à déterminer peut être imposé par le mot-clé NB_FREQ (3 par défaut).

Il est également possible, en utilisant la commande CHAR_CRIT, de choisir la bande dans laquelle il faut chercher ces valeurs propres (de –10 à 10 par défaut).

Remarque

L'indication d'une bande de fréquence est utile surtout pour des calculs en petites perturbations où un test de Sturm est effectué pour la bande de fréquences fournie. On peut ainsi gagner du temps en ne calculant les valeurs propres que s'il y en a dans la bande indiquée. Le test de Sturm n'est pas fait en grandes déformations et les valeurs propres sont calculées à chaque pas de temps.

Le mode de flambement ainsi que les valeurs propres déterminés peuvent être récupérés en utilisant la commande IMPR_RESU :

```
IMPR_RESU( MODELE = MODELE ,FORMAT = 'RESULTAT' ,RESU=(_F(RESULTAT=RESU,
                    NOM_PARA='CHAR_CRIT','MODE_FLAMB',),),)
```

Le cas-test SSLL105 [V3.01.105] propose un exemple d'utilisation de ce critère de stabilité pour un cas linéaire et le cas-test SSNL126 [V6.02.126] pour un cas non linéaire (poutre élastoplastique).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.6 Pilotage du chargement

Afin de faciliter la convergence du calcul incrémental lorsque l'on est proche du niveau de charge ultime, ou afin de pouvoir dépasser ce point critique, il peut être judicieux de ne plus se placer en chargement imposé pour privilégier un pilotage en déplacement ou un pilotage par longueur d'arc. Le pilotage ne peut être utilisé avec le contact [U4.51.03].

4 Approche transitoire couplée

4.1 Problème de référence

On sort ici du cadre réglementaire et l'on va exploiter toutes les possibilités de modélisation offertes par le *Code_Aster*. Le modèle du réservoir lui-même reste inchangé (coques volumiques élastoplastiques). En revanche, on va représenter le domaine fluide par un maillage massif. De plus, la résolution se fera en dynamique transitoire avec l'opérateur DYNA_NON_LINE ([U4.53.01] , [R5.05.05]), la sollicitation externe étant du type sismique.
Le domaine fluide est modélisé en acoustique linéaire (barotrope, compressible, non visqueux et avec surface libre). Le problème couplé fluide-structure est résolu dans le *Code_Aster* par une formulation symétrique (u, p, ϕ) ([R4.02.02], [bib7]), en écriture lagrangienne réactualisée. Le chargement est du type accélérogramme imposé en base de bâche.
Le problème discrétisé se présente alors ainsi [bib6] :

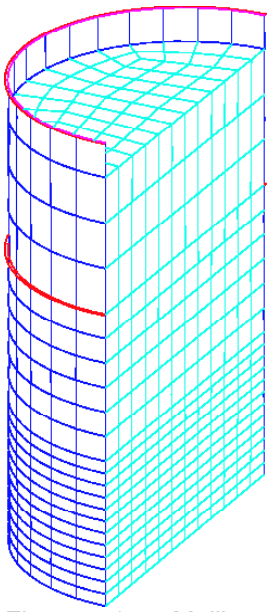

Figure 4.1-a : Maillage complet avec le domaine fluide et la structure

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

4.2 Modélisation couplée fluide-structure dans le *Code_Aster*

Afin de pouvoir mener un calcul couplé par formulation (u, p, ϕ) [bib7] dans le *Code_Aster* , avec surfaces libres et glissement à l'interface (non adhérence entre le fluide non visqueux et la paroi interne du réservoir), il faut respecter une certaine construction du maillage et des modèles correspondants.
On doit donc définir le maillage fluide, le maillage structure et l'interface fluide-structure.
Dans le mailleur on génère donc deux maillages distincts (mailles et nœuds différents pour permettre le glissement à l'interface fluide-structure) pour le domaine fluide et la structure.
Ensuite, dans le *Code_Aster*, on va générer le maillage support de l'interface comme suit, à partir des groupes de mailles SURF0 , SURF1 et SURF2 qui sont les viroles du réservoir :

```
MAILLA01=CREA_MAILLAGE(MAILLAGE=MAILLA1,
                       CREA_GROUP_MA=_F(NOM='IFLUSTRU',
                                        GROUP_MA=('SURF0','SURF1','SURF2',),
                                  PREF_MAILLE='I',),);
```

On crée donc un nouveau groupe de mailles IFLUSTRU . Les nœuds de ce maillage sont confondus avec ceux de la structure, mais les mailles sont différentes.

Remarques

La structure est maillée en éléments COQUE_3D , dont le support géométrique est un quadrilatère à 9 nœuds. Le nœud milieu présente la particularité de ne porter que des ddl de rotation.
Si l'on veut coupler un tel élément avec un élément massif de fluide, on ne peut donc écrire de condition cinématique de couplage pour ce nœud milieu puisqu'il ne comporte pas de ddl de translation, contrairement au nœud correspondant qui vient du domaine fluide et qui, lui, ne porte que des ddl de translation.
Pour contourner ce problème, on ne va écrire le couplage fluide-structure que sur les nœuds du maillage coque comportant les ddl de translation : les nœuds sommets et les nœuds milieux des arêtes des éléments.
Il faut donc avoir dans le Code_Aster le maillage structure ne comportant que des éléments quadrilatères à 8 nœuds (ce sont bien les nœuds sommets et les nœuds milieux des arêtes), sur lequel on définit l'interface. Le maillage structure pour les COQUE_3D étant défini seulement après, à partir de ce maillage, en rajoutant les nœuds milieux.
Le maillage fluide, pour être conforme avec l'interface, sera lui composé de parallélépipèdes à 20 nœuds.

En outre, même si l'on utilisait un maillage structure avec des éléments de coque dont le support géométrique serait un quadrilatère à 9 nœuds, et où tous les nœuds, même le central portaient des ddl de translation, le couplage fluide-structure pourrait poser un problème. En effet, le maillage fluide massif conforme devrait être réalisé avec des éléments massifs parallélépipédiques à 27 nœuds. Or, certains mailleurs n'offrent pas ce type d'éléments complets qui sont assez peu employés en calcul de structure, contrairement aux éléments quadratiques classiques que sont les parallélépipèdes à 20 nœuds.

Une fois la définition de l'interface, qui est donc maillée en éléments quadrilatères à 8 nœuds (qui sont équivalents aux faces des éléments massifs employés pour le domaine fluide : éléments parallélépipède à 20 nœuds), on peut donc faire la modification du maillage structure (groupe de mailles RESERVOI) en éléments à 9 nœuds, support géométrique des COQUE_3D :

```
MAILLA2=CREA_MAILLAGE(MAILLAGE=MAILLA01,
                      MODI_MAILLE=_F(GROUP_MA='RESERVOIR',
                                     OPTION='QUAD8_9',
                                     PREF_NOEUD='NSQ',
                                     PREF_NUME=1,),);
```

On peut alors définir les modèles nécessaires au calcul couplé :

```
MODELE=AFFE_MODELE(MAILLAGE=MAILLA2, INFO=1, VERIF='MAILLE',
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                    AFFE=(_F(GROUP_MA='SURFLIBR',
                             PHENOMENE='MECANIQUE',
                             MODELISATION='2D_FLUI_PESA',),
                          _F(GROUP_MA=('FLUID0','FOND','PLANCENT',),
                             PHENOMENE='MECANIQUE',
                             MODELISATION='3D_FLUIDE',),
                          _F(GROUP_MA='IFLUSTRU',
                             PHENOMENE='MECANIQUE',
                             MODELISATION='FLUI_STRU',),
                          _F(GROUP_MA='RESERVOI',
                             PHENOMENE='MECANIQUE',
                             MODELISATION='COQUE_3D',),),);
```

Le groupe de mailles SURFLIBR (mailles de bord du domaine fluide, situées sur sa face supérieure) porte un modèle de surface libre [R4.02.04] : 2D_FLUID_PESA .
Les groupes de mailles FLUID0 (domaine fluide massif), FOND (mailles de bord de FLUID0 définissant le fond) et PLANCENT (mailles de bord de FLUID0 définissant le plan de symétrie) définissent le domaine fluide total : 3D_FLUIDE .
L'interface fluide-structure (FLUI_STRU) est portée par le groupe de mailles IFLUSTRU .
Enfin, le maillage du réservoir (RESERVOI) est le support du modèle structure en coques volumiques (COQUE_3D).

On définit aussi le matériau compressible fluide :

```
EAU=DEFI_MATERIAU(FLUIDE=_F(RHO=1000.0, CELE_R=1500.0,),);
```

que l'on affectera au domaine fluide (modèle massif et ses bords) ainsi qu'à l'interface IFLUSTRU et à la surface libre SURFLIBR :

```
       CHMAT=AFFE_MATERIAU(MAILLAGE=MAILLA2,
             AFFE=(_F(GROUP_MA=('FLUID0','FOND','PLANCENT','IFLUSTRU','SURFLIBR',),
            MATER=EAU,),
…
                        ) ;
```

4.3 Conditions aux limites

Les conditions aux limites cinématiques portent sur l'encastrement en base de réservoir (SYMETRI2), sur les génératrices dans le plan vertical de symétrie (SYMETRIE) et sur la non pénétration (vitesse normale nulle) au fond du domaine fluide (FOND), ainsi que dans son plan vertical de symétrie (PLANCENT).

```
       CONDLIM=AFFE_CHAR_MECA(MODELE=MODELE,
                         DDL_IMPO=(_F(GROUP_NO='SYMETRI2',
                                      DX=0.0, DY=0.0, DZ=0.0,
                                      DRX=0.0, DRY=0.0, DRZ =0.0,),
                                   _F(GROUP_NO='SYMETRIE',
                                      DY=0.0, DRX=0.0, DRZ=0.0,),),
                         VITE_FACE=_F(GROUP_MA=('FOND','PLANCENT',), VNOR=0.0,),);
```

Les effets de pesanteur sont pris en compte :

```
       PESA=AFFE_CHAR_MECA(MODELE=MODELE, PESANTEUR=(9.81,0.0,0.0,-1.0,),);
```

Cependant cette commande est insuffisante car dans la modélisation (u, p, ϕ) [bib7] les effets de pesanteur dans le domaine fluide ne peuvent être pris en compte. Si on ne faisait rien de plus, la pesanteur ne serait donc véritablement imposée que sur la structure.
Pour approcher les effets de pesanteur du fluide sur la paroi, on va imposer une pression hydrostatique équivalente (mais qui ne peut tenir compte de la variation de hauteur du fluide au cours du calcul quand le ballottement va commencer) :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
       PHZ=DEFI_FONCTION(NOM_PARA='Z',
       …
                         VERIF='CROISSANT',);
       #
       PH = FORMULE(NOM_PARA=('X','Y','Z'),VALE='PHZ(Z)',),
       #
       PRESSHYD=AFFE_CHAR_MECA_F(MODELE=MODELE,
                                 FORCE_COQUE=_F(GROUP_MA='VIROLE', PRES= PH ,
                                        PLAN='INF',),);
```

Le séisme est imposé comme étant un accélérogramme (fonction GASDM_X1) imposé en base suivant la direction X. C'est donc une sollicitation de type mono-appui classique :

```
       ACCELERX=CALC_FONCTION(COMB=(_F(FONCTION=GASDM_X1, COEF=0.5,),),);
       #
       MULT_X=CALC_CHAR_SEISME(MATR_MASS=MATMAS,
                               DIRECTION=(1.0,0.0,0.0,), _APPUI='OUI',);
       #
CHARG_SE=AFFE_CHAR_MECA(MODELE=MODELE, VECT_ASSE=MULT_X,);
```

La matrice de masse utilisée (MATMAS) est la matrice de masse du système couplé total.

4.4 Conditions initiales

L'état initial du calcul transitoire doit correspondre à l'équilibre du système total lorsqu'il n'est pas soumis au séisme. Cet état d'équilibre statique correspond donc au chargement de pesanteur et aux effets hydrostatiques. Si l'on commençait le calcul dynamique avec un état initial ne respectant pas cet équilibre, alors cela génèrerait des oscillations de la solution transitoire puisqu'elle ne serait pas initialement à l'équilibre (le niveau sismique étant cependant alors nul). On peut atténuer ces oscillations en ajoutant un amortissement structural « grand » et en attendant que la solution se stabilise avant d'imposer le séisme, mais cette technique est assez peu élégante…
Pour calculer cet état initial statiquement équilibré, on peut donc résoudre un problème statique (que l'on suppose en plus linéaire) d'équilibre sous l'action des forces de pesanteur et hydrostatiques.
Pour ce faire, on calcule et assemble préalablement les matrices globales **K** et **M** avec le chargement hydrostatique et la pesanteur :

```
MACRO_MATR_ASSE( MODELE = MODELE, CHAM_MATER=CHMAT,
                 CARA_ELEM=CARAELEM, CHARGE     = (CONDLIM,PESA,PRESSHYD,),
                 NUME_DDL   = CO('NUMSTA'),
                 SOLVEUR=_F(METHODE='MULT_FRONT',RENUM='METIS'),
                 MATR_ASSE=(
                  _F( MATRICE= CO('RIGSTA'), OPTION= 'RIGI_MECA'),
                  _F( MATRICE= CO('MASSTA'), OPTION= 'MASS_MECA'), ),);
```

La matrice de raideur assemblée étant singulière à cause du domaine fluide (la formulation (u, p, ϕ) rend cette matrice singulière à fréquence nulle [bib8]), on modifie légèrement le problème en considérant la matrice de raideur $\mathbf{K}_{cor} = \mathbf{K} + \varepsilon \mathbf{M}, \quad \varepsilon << 1$ qui n'est plus singulière (on la nomme RIGICOMB).
On peut, par exemple, prendre ε = 0,001 comme ci-dessous :

```
RIGICOMB = COMB_MATR_ASSE(COMB_R=(_F( MATR_ASSE= RIGSTA, COEF_R= 1.),
                                  _F( MATR_ASSE= MASSTA, COEF_R= -0.001),);
```

On assemble ensuite le vecteur chargement F_0 (second membre) :

```
       E_0 = CALC_VECT_ELEM(CARA_ELEM=CARAELEM, CHAM_MATER=CHMAT,
                            OPTION='CHAR_MECA',CHARGE=(CONDLIM, PESA,PRESSHYD,),);
       F_0      = ASSE_VECTEUR(VECT_ELEM= E_0, NUME_DDL= NUMSTA );
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

On peut alors résoudre le problème de statique linéaire $\mathbf{K}_{cor}\underline{U}=\underline{F}_0$, en utilisant, par exemple, une factorisation de type LDL^T :

```
RIGICOMB = FACTORISER(reuse=RIGICOMB, MATR_ASSE= RIGICOMB,
                      STOP_SINGULIER= 'NON');
DEP0     = RESOUDRE(MATR_FACT= RIGICOMB, CHAM_NO=F_0);
```

Le champ solution DEP0 sera donc l'état initial du calcul dynamique transitoire qui suit.

4.5 Résolution transitoire

On utilise l'opérateur DYNA_NON_LINE ([U4.53.01] , [R5.05.05]) comme suit :

```
RESU=DYNA_NON_LINE(MODELE=MODELE,
                   CHAM_MATER=CHMAT,
                   CARA_ELEM=CARAELEM,
                   EXCIT=(_F(CHARGE=CONDLIM,),
                          _F(CHARGE=PESA,),
                          _F(CHARGE=PRESSHYD,
                             FONC_MULT=FONCMUL0,
                             TYPE_CHARGE='SUIV',),
                          _F(CHARGE=CHARG_SE,
                             FONC_MULT=ACCELERX,),),
                   COMP_INCR=(_F(RELATION='ELAS',
                             DEFORMATION='PETIT_REAC',
                             GROUP_MA=('FLUID0','FOND','PLANCENT',
                                       'IFLUSTRU','SURFLIBR',),),
                              _F(RELATION='ELAS', DEFORMATION='PETIT_REAC',
                                 GROUP_MA='ANNEAU',),
                              _F(RELATION='VMIS_ISOT_TRAC',
                                 DEFORMATION='PETIT_REAC',
                                 GROUP_MA=('SURF0','SURF1',
                                           'SURF2','SURF3',),),),
                   ETAT_INIT=_F(INST_ETAT_INIT=0.0, DEPL=DEP0,),
                   INCREMENT=_F(LIST_INST=LINST,
                                SUBD_PAS=4,
                                SUBD_PAS_MINI=1e-06,
                                SUBD_METHODE='UNIFORME', ),
                   SCHEMA_TEMPS=_F(SCHEMA='HHT', ALPHA=-0.1,
                                   FORMULATION='DEPLACEMENT',),
                   NEWTON=_F(REAC_INCR=1, MATRICE='TANGENTE',REAC_ITER=1,),
                   SOLVEUR=_F(STOP_SINGULIER='NON',),
                   CONVERGENCE=_F(RESI_GLOB_RELA=1.e-05, ITER_GLOB_MAXI=20,
                                  ARRET='OUI',),
                   ARCHIVAGE=_F(LIST_INST=LARCH, ARCH_ETAT_INIT='OUI',),);
```

La résolution se fait en lagrangien réactualisé (option DEFORMATION='PETIT_REAC') car le domaine fluide est en petites perturbations sur chaque pas. Il faut donc vérifier que le pas de temps est suffisamment petit pour que cette hypothèse soit vérifiée.
On utilise un schéma d'intégration en temps de type accélération moyenne modifiée (SCHEMA='HHT', ALPHA=-0.1) avec amortissement numérique afin de stabiliser la solution et de faciliter la convergence.

Le cas-test FDNV100 [V8.03.100] présente le calcul d'une cuve rectangulaire pleine d'eau avec une paroi souple. La modélisation mise en œuvre est très proche de celle utilisée ici pour les grands réservoirs.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5 Bibliographie

1) S. Goubet : *Conception / vérification sismique des grands réservoirs à fond plat*. Note EDF SEPTEN ENGSDS030109.

2) N. Greffet, A. Assire, J.Pigat, J.-M. Proix : *Etude prototype de réservoirs métalliques au séisme - Analyse de stabilité non linéaire*. Note EDF R&D HT-62/02/011/B.

3) N. GREFFET : *Etude numérique d'une maquette de bâche : comparaison avec les essais Freyssinet*. Note EDF R&D HT-62/03/009/A.

4) N. GREFFET : *Etude prototype de réservoirs métalliques au séisme - Analyse de stabilité non linéaire*, Actes du VI^{e} Colloque national AFPS, vol. II, p. 225-232, 1-3 juillet, 2003.

5) N. GREFFET : *Modélisation du décollement pour la simulation d'une bâche sous chargement sismique réglementaire*. Note EDF R&D HT-62/04/026/A.

6) N. GREFFET : *Instabilité non linéaire de grands réservoirs sous séisme - De l'approche réglementaire vers le transitoire couplé fluide-structure*, Actes du 7^{e} Colloque National en Calcul des Structures, Vol. 1, p. 265-270, Giens, 17-20 Mai, 2005

7) H.J.-P. Morand, R. Ohayon : *Interactions fluides-structures*, Editions Masson, 1992.

8) N. Greffet : V*oies d'amélioration de la formulation couplée fluide-structure dans Code_Aster*, Note EDF R&D HT-62/02/023/A.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)